IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Appellants:	Fabio CASATI et al.	§ §	Confirmation No.:	6697
Serial No.:	10/681,782	§ §	Group Art Unit:	2445
Filed:	October 8, 2003	§ §	Examiner:	Thomas Duong
For:	A Method And System For Business-Oriented Web Services Management	§ § §	Docket No.:	200311728-1

APPEAL BRIEF

Mail Stop Appeal Brief – Patents October 18, 2010
Commissioner for Patents
PO Box 1450
Alexandria, VA 22313-1450

Sir or Madam,

Appellants hereby submit this Appeal Brief in connection with the above-identified application. A Notice of Appeal was electronically filed on August 16, 2010.

TABLE OF CONTENTS

I. REAL PARTY IN INTEREST 3
II. RELATED APPEALS AND INTERFERENCES 4
III. STATUS OF THE CLAIMS 5
IV. STATUS OF THE AMENDMENTS 6
V. SUMMARY OF THE CLAIMED SUBJECT MATTER 7
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL 10
VII. ARGUMENT 11
  A. Discussion of the Cited Art 11
    1. Pagnano (U.S. 7,110,843) 11
    2. Murstein (U.S. 6,789,046) 11
    3. Fraenkel (U.S. 2002/0198985) 12
  B. Rejections Under 35 U.S.C. § 103(a) of Claims 1, 4–11, and 13–30—Fraenkel in View of Murstein in View of Curley in View of Pagnano 12
    1. Claims 1, 4–11, 13–21, and 29–30 12
    2. Claims 22–24 13
    3. Claims 25–26 14
    4. Claims 27–28 15
  C. Conclusion 16
VIII. CLAIMS APPENDIX 17
IX. EVIDENCE APPENDIX 25
X. RELATED PROCEEDINGS APPENDIX 26

I. REAL PARTY IN INTEREST

The real party in interest is Hewlett-Packard Development Company, LP, a limited partnership established under the laws of the State of Texas and having a principal place of business at 11445 Compaq Center Drive West, Houston, Texas, 77070, U.S.A. (hereinafter "HPDC"). HPDC is a Texas limited partnership and is a wholly-owned affiliate of Hewlett-Packard Company, a Delaware Corporation, headquartered in Palo Alto, CA. The general or managing partner of HPDC is HPQ Holdings, LLC.

II. RELATED APPEALS AND INTERFERENCES

Appellants are unaware of any related appeals or interferences.

III. STATUS OF THE CLAIMS

Originally filed claims: 1-30.

Claim cancellations: 2-3 and 12.

Added claims: None.

Presently pending claims: 1, 4-11 and 13-30.

Presently appealed claims: 1, 4-11 and 13-30.

IV. STATUS OF THE AMENDMENTS

No claims were amended after the Final Office Action dated June 14, 2010.

V. SUMMARY OF THE CLAIMED SUBJECT MATTER

This section provides a concise explanation of the subject matter defined in each of the independent claims, referring to the specification by page and line number or to the drawings by reference characters as required by 37 C.F.R. § 41.37(c)(1)(v). Each element of the claims is identified with a corresponding reference to the specification or drawings where applicable. The specification references are made to the application as filed by Appellants. Note that the citation to passages in the specification or drawings for each claim element does not imply that the limitations from the specification and drawings should be read into the corresponding claim element. Also note that these specific references are not exclusive; there may be additional support for the subject matter elsewhere in the specification and drawings.

Claim 1 recites a method for monitoring performance of web-based services, and the method includes receiving a transaction from a client, fig.1 ref.110; p.2 l.32; p.4 l.31; p.5 ll.1–2. The method further includes routing the transaction to an appropriate web service for execution of the transaction, fig.1 refs.120, 122, 124; fig.2 ref.250; p.3 l.1; p.5 ll.21–23. The method further includes executing the transaction, p.3 l.2; p.5 l.30–p.6 l.1, and monitoring and logging transaction execution performance data relating to receiving and executing the transaction, p.3 l.3; p.6 l.1. The method further includes reporting the performance data based on pre-defined performance metrics, p.3 l.5; p.7 ll.6–7; p.9 ll.1–2. Receiving the transaction includes tracking, via a Simple Object Access Protocol ("SOAP") tracker, of information relating to the transaction, fig.2 ref.210; p.5 l.19; p.6 ll.19–22. Receiving the transaction also includes communicating, via the SOAP tracker, the information to a web services manager (WSM) for performance analysis, fig.2 refs.210, 230; p.6 ll.9–23.

Claim 11 recites a computer system for monitoring performance of web-based services, and the computer system includes a tracking system for tracking information relating to a transaction communicated from a client, p.3 l.3; p.6 l.1; fig.2 ref.210; p.5 l.19; p.6 ll.19–22. The system further includes a routing system for routing the transaction to an appropriate web service for execution of the transaction, fig.1 refs.120, 122, 124; fig.2 ref.250; p.3 l.1; p.5 ll.21–23. The system further includes a data collection system for collecting performance data relating to the communication of the transaction from the client and the execution of the transaction by the web service, p.3 l.2; p.5 l.30–p.6 l.1; p.3 l.3; p.6 l.1. The system further includes a reporting system for extracting the performance data and generating reports relating to the performance of the web services based on pre-defined performance metrics, p.3 l.5; p.7 ll.6–7; p.9 ll.1–2. The tracking system includes a Simple Object Access Protocol ("SOAP") tracker, fig.2 ref.210; p.5 l.19; p.6 ll.19–22.

Claim 22 recites a method for creating custom metrics for monitoring performance of web-based services, and the method includes providing a metrics construction model, p.5 ll.6–8. The method further includes providing a metrics library containing metrics definitions, p.6 ll.19–22, and selecting at least one built-in metric function from the metrics library, p.7 l.15, 21, 28; p.8 l.9. The method further includes customizing the selected built-in metric function based on the metrics construction model by selecting parameters for the built-in metric function, p.7 l.28–p.8 l.28; p.7 l.15, 21, 28; p.8 l.9.

Claim 25 recites a method for correlating conversation metrics to composition metrics to analyze performance of web-based services, and the method includes collecting information relating to at least one metric associated with a conversation, p.4 l. 25–p.5 l.8. The method further includes collecting information relating to at least one metric associated with a composition, p.5 l.9–p.6 l.8. The method further includes correlating the collected conversation metric information to the collected composition metric information to determine the relationship between execution of instances of the execution of the composition and instances of execution of the conversation in terms of the performance of the web-based service, p.9 l.18–p.10 l.7.

Claim 27 recites a computing means, p.3 ll.24–26, for managing business-oriented web services, and the computing means, p.3 ll.24–26, includes a client interface means, p.4 ll.17–18, 24, for conducting a conversation with a client. The conversation includes passing at least one transaction between the client and a web service, fig.1 ref.110; p.2 l.32; p.4 l.31; p.5 ll.1–2. The computer means further includes an operation means, p.7 l.5, 16, for executing the transaction based on a composition, wherein the composition includes a business logic internal to the web service, p.5 l.9–p.6 l.8. The computer means further includes a conversation tracking means, p.4 l. 25–p.5 l.8, for monitoring and storing data relating to the conversation performance, p.4 l. 25–p.5 l.8. The computer means further includes a composition tracking means, p.5 l.9–p.6 l.8, for monitoring and storing data relating to the composition performance, p.5 l.9–p.6 l.8. The computer means further includes a processing means, p.6 l.28; p.3 ll.24–26, for compiling stored data relating to the conversation and composition into a report, p.3 l.5; p.7 ll.6–7; p.9 ll.1–2.

Claim 28 recites the computing means of claim 27, wherein a web services manager generates the report for display on a web framework, fig.3 ref.332, p.9 ll.9–14.

Claim 29 recites a computer-readable storage medium for storing an executable set of software instructions, and the instructions are configured for managing business-oriented web services by a method comprising transmitting a transaction from a client to an appropriate web service for execution of the transaction, fig.1 ref.110; p.2 l.32; p.4 l.31; p.5 ll.1–2. The instructions are further configured for executing the transaction, p.3 l.2; p.5 l.30–p.6 l.1. The instructions are further configured for monitoring and logging, via a Simple Object Access Protocol ("SOAP") tracker, transaction execution performance data relating to transmitting and executing the transaction to a service execution log, p.3 l.3; p.6 l.1; fig.2 ref.210; p.5 l.19; p.6 ll.19–22. The instructions are further configured for extracting the performance data from the service execution log, p.6 ll.2–8; p.8 l.7; p.5 l.26. The instructions are further configured for parsing the performance data into a format readable by a web services manager, p.5 l.26. The instructions are further configured for compiling the performance data into a report, p.3 l.5; p.7 ll.6–7; p.9 ll.1–2.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Whether claims 1, 4-11 and 13-30 are obvious due to Fraenkel (U.S. Pat. Pub. No. 2002/0198985) in view of Murstein (U.S. Pat. No. 6,789,046), Curley (U.S. Pat. Pub. No. 2002/0120727) and Pagnano (U.S. Pat. No. 7,110,843).

VII. ARGUMENT

The claims do not stand or fall together. Instead, Appellants present separate arguments for various independent and dependent claims. After a brief discussion of the cited art, each of these arguments is separately argued below and presented with separate headings as required by 37 C.F.R. § 41.37(c)(1)(vii).

A. Discussion of the Cited Art

1. Pagnano (U.S. 7,110,843)

At col. 2, ll. 29–46, Pagnano discloses

> This and other advantages can be achieved with an exemplary embodiment of the arrangement and method according to the present invention. In particular, a first processing system can access monitoring data from a database of a second processing system using a Web Service, in which the monitoring data is associated with a field device (e.g., a temperature sensor, a pressure sensor, a flow rate sensor, a valve, and/or a switch). For example, the monitoring data can be in an Extensible Markup Language ("XML") format, and the first processing system can use a Simple Object Access Protocol ("SOAP") to access a particular portion of the monitoring data from the database. Moreover, the first processing system may transmit a request to replace the field device and/or a recommendation for maintenance to be performed on the field device based on the monitoring data.

2. Murstein (U.S. 6,789,046)

Claim 23 of Murstein discloses

> 23. A method for monitoring performance metrics of a plurality of members configured as an entity, comprising: setting configurable performance metric settings for determining performance metric types to be monitored; propagating the performance metric settings to a plurality of remaining members of the entity to establish performance metric configuration settings at the plurality of members; logging performance metric data at predefined time periods and resolutions at each of the plurality of members based on the performance metric configuration settings at each of the plurality of members; and dynamically updating the logging of performance metric data at each of the plurality of members based on any changes in the performance metric settings.

3. Fraenkel (U.S. 2002/0198985)

At ¶ 0228, Fraenkel discloses

> Users preferably connect to the RCA system 168 through the RCA Internet server 268 . The RCA Internet server 268 includes a RCA controller 274 that receives requests from the user devices, such as the browser 100 , and forwards the requests to the appropriate RCA system component. For example, in one embodiment, the RCA controller 274 is a servlet, such as Tomcat or another Java servlet, capable of receiving Hypertext Transfer Protocol (HTTP) requests from the browser 100 and converting the requests to inner structures and method calls. The RCA Internet server 268 also includes a document object 276 and a user interface component 278.

B. Rejections Under 35 U.S.C. § 103(a) of Claims 1, 4–11, and 13–30—Fraenkel in View of Murstein in View of Curley in View of Pagnano

1. Claims 1, 4–11, 13–21, and 29–30

Independent claim 1, recites, in part, "wherein receiving the transaction includes tracking, by a Simple Object Access Protocol ("SOAP") tracker, of information relating to the transaction; wherein receiving the transaction includes communicating, by the SOAP tracker, the information to a web services manager (WSM) for performance analysis." Independent claims 11 and 29 recite similar limitations. However, Examiner erred in rejecting the claims because the cited combination fails to teach or suggest the quoted limitations. On page 6 of the Final Office Action, Examiner notes that Fraenkel, Murstein, and Curley do not teach the quoted limitations. Examiner cites Pagnano as allegedly teaching the quoted limitation. Pagnano states:

> This and other advantages can be achieved with an exemplary embodiment of the arrangement and method according to the present invention. In particular, **a first processing system can access monitoring data from a database** of a second processing system using a Web Service, in which the monitoring data is associated with a field device (e.g., a temperature sensor, a pressure sensor, a flow rate sensor, a valve, and/or a switch). For example, the monitoring data can be in an Extensible Markup Language ("XML") format, and **the first processing system can use a Simple Object Access Protocol ("SOAP") to access a particular portion of the monitoring data from the database.** Moreover, the first processing system may transmit a request to replace the field device

> and/or a recommendation for maintenance to be performed on the field device based on the monitoring data.

Col. 2, ll. 29–46 (emphasis added).

However, using SOAP to access a particular portion of the database of a field device that tracks, e.g., temperature, pressure, and flow rate fails to teach or suggest a SOAP tracker performing the tracking. Specifically, accessing a database fails to teach or suggest tracking information relating to a transaction because the two operations are distinct. The mere commonality of using SOAP does not transform the former into teaching or suggesting the latter. Furthermore, no other art of record teaches or suggests the limitations as required by the claims. For at least this reason, Examiner erred in rejecting independent claims 1, 11, and 29, along with dependent claims 4–10, 13–21, and 30, and the rejections should be reversed.

2. Claims 22–24

Independent claim 22 recites, in part, "providing a metrics library containing metrics definitions." However, Examiner erred in rejecting the claims because the cited combination fails to teach or suggest the quoted limitation. Examiner cites Murstein at claim 23 as allegedly teaching the quoted limitation. At the cited location, Murstein states

> 23. A method for monitoring performance metrics of a plurality of members configured as an entity, comprising: **setting configurable performance metric settings for determining performance metric types to be monitored;** propagating the performance metric settings to a plurality of remaining members of the entity to establish performance metric configuration settings at the plurality of members; logging performance metric data at predefined time periods and resolutions at each of the plurality of members based on the performance metric configuration settings at each of the plurality of members; and dynamically updating the logging of performance metric data at each of the plurality of members based on any changes in the performance metric settings.

(Emphasis added.)

However, setting configurable performance metric settings for determining performance metric types to be monitored fails to teach or suggest providing a metrics library containing metrics definitions. Specifically, choosing a setting for a variable fails to teach or suggest providing the variable and the definition of the variable among a library of variable definitions because the input considered and actions taken while performing the former are different from performing the latter. Furthermore, no other art of record teaches or suggests the limitations as required by the claims. For at least this reason, Examiner erred in rejecting independent claim 22, along with dependent claims 23–24, and the rejections should be reversed.

3. Claims 25–26

Independent claim 25 recites, in part, "correlating the collected conversation metric information to the collected composition metric information to determine the relationship between execution of instances of the execution of the composition and instances of execution of the conversation in terms of the performance of the web-based service." However, Examiner erred in rejecting the claims because the cited combination fails to teach or suggest the quoted limitation. Examiner cites Murstein at claim 23 as allegedly teaching the quoted limitation. At the cited location, Murstein states

> 23. A method for monitoring performance metrics of a plurality of members configured as an entity, comprising: setting configurable performance metric settings for determining performance metric types to be monitored; propagating the performance metric settings to a plurality of remaining members of the entity to establish performance metric configuration settings at the plurality of members; logging performance metric data at predefined time periods and resolutions at each of the plurality of members based on the performance metric configuration settings at each of the plurality of members; and **dynamically updating the logging of performance metric data at each of the plurality of members based on any changes in the performance metric settings**.

(Emphasis added.)

However, dynamically updating the logging of performance metric data based on any changes in the performance metric settings fails to teach or suggest correlating the collected conversation metric information to the collected composition metric information. Specifically, changing how a variable is logged fails to teach or suggest correlation of a conversation variable and a composition variable because the former requires changing how one variable is recorded while the latter requires measuring how closely at least two variables are related. Furthermore, no other art of record teaches or suggests the limitations as required by the claims. For at least this reason, Examiner erred in rejecting independent claim 25, along with dependent claim 26, and the rejections should be reversed.

4. Claims 27–28

Independent claim 27 recites, in part, "a conversation tracking means for monitoring and storing data relating to the conversation performance; a composition tracking means for monitoring and storing data relating to the composition performance." However, Examiner erred in rejecting the claims because the cited combination fails to teach or suggest the quoted limitation. Examiner cites Fraenkel at ¶ 0228 as allegedly teaching the quoted limitation. At the cited location, Fraenkel states

> Users preferably connect to the RCA system 168 through the RCA Internet server 268 . The RCA Internet server 268 includes a RCA controller 274 that receives requests from the user devices, such as the browser 100 , and forwards the requests to the appropriate RCA system component. For example, in one embodiment, the RCA controller 274 is a servlet, such as Tomcat or another Java servlet, capable of receiving Hypertext Transfer Protocol (HTTP) requests from the browser 100 and converting the requests to inner structures and method calls. The RCA Internet server 268 also includes a document object 276 and a user interface component 278.

However, the RCA system of Fraenkel fails to teach or suggest conversation tracking and composition tracking. Specifically, Fraenkel fails to make the distinction between conversation tracking and composition tracking. Furthermore, no other art of record teaches or suggests the limitations as required by the claims. For at least this reason, Examiner erred in rejecting independent claim 27, along with dependent claim 28, and the rejections should be reversed.

C. Conclusion

For the reasons stated above, Appellants respectfully submit that the rejections should be reversed. In the course of the foregoing discussions, Appellants may have at times referred to claim limitations in shorthand fashion, or may have focused on a particular claim element. This discussion should not be interpreted to mean that the other limitations can be ignored or dismissed, or that limitations from the specification can be imported into the claims. The claims must be viewed as a whole, and each limitation of the claims must be considered when determining the patentability of the claims. Moreover, it should be understood that there may be other distinctions between the claims and the prior art which have yet to be raised, but which may be raised in the future.

It is believed that no extensions of time or fees are required, beyond those that may otherwise be provided for in attachments accompanying this document. However, in the event that additional extensions of time are necessary to allow consideration of this document, such extensions are hereby petitioned under 37 C.F.R. § 1.136(a), and any fees required (including fees for net addition of claims) are hereby authorized to be charged to Hewlett-Packard Development Company's Deposit Account No. 08-2025.

Respectfully submitted,

/Tim D. Chheda/

Tim D. Chheda
PTO Reg. No. 60,752
CONLEY ROSE, P.C.
(713) 238-8000 (Phone)
(713) 238-8008 (Fax)
ATTORNEY FOR APPELLANTS

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
Legal Dept., M/S 35
3404 E. Harmony Road
Fort Collins, CO 80528

VIII. CLAIMS APPENDIX

1. A method for monitoring performance of web-based services, the method comprising:

receiving a transaction from a client;

routing the transaction to an appropriate web service for execution of the transaction;

executing the transaction;

monitoring and logging transaction execution performance data relating to receiving and executing the transaction; and

reporting the performance data based on pre-defined performance metrics;

wherein receiving the transaction includes tracking, via a Simple Object Access Protocol ("SOAP") tracker, of information relating to the transaction;

wherein receiving the transaction includes communicating, via the SOAP tracker, the information to a web services manager (WSM) for performance analysis.

4. The method of claim 1, wherein the routing step is performed by a SOAP router or application server.

5. The method of claim 1, wherein the monitoring and logging step includes the use of a service registry and a service composition engine to record and store

data in service execution logs relating to instances of conversation transaction or composition transaction.

6. The method of claim 1, wherein the reporting step is performed by the WSM.

7. The method of claim 1, wherein the reporting step includes extracting data from a service registry or the service execution logs for use by the WSM to generate performance reports.

8. The method of claim 6, wherein the WSM further includes a WSM definer, a metric schema and a WSM engine to generate performance reports.

9. The method of claim 8, wherein the WSM engine receives data from the SOAP tracker and data relating to conversation definitions, composition service definitions and related execution data.

10. The method of claim 6, wherein the WSM engine communicates with a web framework and related application servers to provide performance reports to business or IT personnel.

11. A computer system for monitoring performance of web-based services, the computer system comprising:

a tracking system for tracking information relating to a transaction communicated from a client;

a routing system for routing the transaction to an appropriate web service for execution of the transaction;

a data collection system for collecting performance data relating to the communication of the transaction from the client and the execution of the transaction by the web service; and

a reporting system for extracting the performance data and generating reports relating to the performance of the web services based on pre-defined performance metrics;

wherein the tracking system further comprises a Simple Object Access Protocol ("SOAP") tracker.

13. A system as recited in claim 11, wherein the tracking system receives tracking information relating to the transaction and communicates the information to a web services manager (WSM).

14. A system as recited in claim 11, wherein the routing system further comprises a SOAP router or application server.

15. A system as recited in claim 11, wherein the data collection system further comprises a service registry and a service composition engine to memorialize web service performance data in service execution logs.

16. A system as recited in claim 15, wherein the web service performance data in the service execution logs comprises data relating to instances of conversation transaction or composition transaction.

17. A system as recited in claim 11, wherein the reporting system further comprises a WSM.

18. A system as recited in claim 11, wherein the reporting system receives data from a service registry or service execution logs and transmits the data to the WSM.

19. A system as recited in claim 18, wherein the WSM comprises a WSM definer, a metric schema and a WSM engine.

20. A system as recited in claim 19, wherein the WSM engine receives data from the SOAP tracker and data relating to conversation definitions, composition service definitions and related execution data.

21. A system as recited in claim 19, wherein the WSM engine communicates with a web framework and related application servers.

22. A method for creating custom metrics for monitoring performance of web-based services, the method comprising:

providing a metrics construction model;

providing a metrics library containing metrics definitions;

selecting at least one built-in metric function from the metrics library; and

customizing the selected built-in metric function based on the metrics construction model by selecting parameters for the built-in metric function.

23. The method of claim 22, wherein the metrics construction model includes an algorithm for assembling the selected at least one metric function and the selected parameters into a custom metric.

24. The method of claim 22, wherein the metrics library is included in the WSM.

25. A method for correlating conversation metrics to composition metrics to analyze performance of web-based services, the method comprising:

collecting information relating to at least one metric associated with a conversation;

collecting information relating to at least one metric associated with a composition; and

correlating the collected conversation metric information to the collected composition metric information to determine the relationship between execution of instances of the execution of the composition and instances of execution of the conversation in terms of the performance of the web-based service.

26. The method of claim 25, wherein conversation metric information is obtained from a service registry; and wherein composition metric information is obtained from a service composition engine.

27. A computing means for managing business-oriented web services, the computing means comprising:

a client interface means for conducting a conversation with a client, wherein the conversation includes passing at least one transaction between the client and a web service;

an operation means for executing the transaction based on a composition, wherein the composition includes a business logic internal to the web service;

a conversation tracking means for monitoring and storing data relating to the conversation performance;

a composition tracking means for monitoring and storing data relating to the composition performance; and

a processing means for compiling stored data relating to the conversation and composition into a report.

28. The computing means of claim 27, wherein a web services manager generates the report for display on a web framework.

29. A computer-readable storage medium for storing an executable set of software instructions, wherein the instructions are configured for managing business-oriented web services by a method comprising:

transmitting a transaction from a client to an appropriate web service for execution of the transaction;

executing the transaction;

monitoring and logging, via a Simple Object Access Protocol ("SOAP") tracker, transaction execution performance data relating to transmitting and executing the transaction to a service execution log;

extracting the performance data from the service execution log;

parsing the performance data into a format readable by a web services manager; and

compiling the performance data into a report.

30. The computer-readable storage medium of claim 29, wherein the compiling step is coordinated by a web services manager.

IX. EVIDENCE APPENDIX

None.

X. RELATED PROCEEDINGS APPENDIX

None.